

# Results - Zero Shot - Code Llama

## Results - Zero Shot - Code Llama

Created: 2024-02-15 14:10

[Robot LLM tasks](#)

### Task 1

#### Prompt

```
<s>[INST] <<SYS>>\nYou will be provided a summary of a task performed by a robot, and your objective is to express this task as a behavior tree in XML format.\n<</SYS>>\n\nThe behavior tree represents a mobile robot tasked to visit a sequence of locations: ((0,0), (2,3), (4, 7), (5, 11)). The available actions are: "moveTo" [/INST]
```

#### Output

```
[INST] <<SYS>>\nYou will be provided a summary of a task performed by a robot, and your objective is to express this task as a behavior tree in XML format.\n<</SYS>>\n\nThe behavior tree represents a mobile robot tasked to visit a sequence of locations: ((0,0), (2,3), (4, 7), (5, 11)). The available actions are: "moveTo" [/INST] <root main_tree_to_execute="MobileRobot">[subtrees]  <MobileRobot>    <seed>0</seed>    <goal_as_string>0;2;3;4;5;11</goal_as_string>    <initial_pose>0;0;0</initial_pose>  </MobileRobot></subtrees>
```

### Task 2

#### Prompt

```
<s>[INST] <<SYS>>\nYou will be provided a summary of a task performed by a
robot, and your objective is to express this task as a behavior tree in XML
format.\n<</SYS>>\n\nThe behavior tree represents a mobile robot tasked to
visit a sequence of locations: ((0,0), (2,3), (4, 7), (5, 11)). Each
location is associated with a reading: (10, 20, 13, 27). Location with a
reading above 15 must be visited first. The available actions are: "moveTo"
[/INST]
```

## Output

```
[INST] <<SYS>>\nYou will be provided a summary of a task performed by a
robot, and your objective is to express this task as a behavior tree in XML
format.\n<</SYS>>\n\nThe behavior tree represents a mobile robot tasked to
visit a sequence of locations: ((0,0), (2,3), (4, 7), (5, 11)). Each
location is associated with a reading: (10, 20, 13, 27). Location with a
reading above 15 must be visited first. The available actions are: "moveTo"
[/INST] <?xml version="1.0"?>
<root main_tree_to_execute = "MainTree" >
  <BehaviorTree ID="MainTree">
    <Sequence name="root">
      <Mobile name="robot">
        <Visit
          location_list="{destination_list}"
          reading_list="{reading_list}"
          order="first_gte"
          name="robot_visit"
          action="moveTo"
        />
      </Mobile>
    </Sequence>
  </BehaviorTree>

  <TreeNodesModel>
    <SubTree ID="Mobile">
      <input_port name="name"/>
      <input_port name="action"/>
      <input_port name="location_list"/>
      <input_port name="reading_list"/>
      <input_port name="order"/>
```

```

        </SubTree>
        <SubTree ID="Visit">
            <input_port name="name"/>
            <input_port name="action"/>
            <input_port name="location_list"/>
            <input_port name="reading_list"/>
            <input_port name="order"/>
        </SubTree>
    </TreeNodeModel>
</root>

```

## Task 3

### Prompt

```

<s>[INST] <<SYS>>\nYou will be provided a summary of a task performed by a
robot, and your objective is to express this task as a behavior tree in XML
format.\n<</SYS>>\n\nThe behavior tree represents a mobile robot tasked to
visit a sequence of locations: ((0,0), (2,3), (4, 7), (5, 11)). During
navigation a location may become unreachable, if this happens, skip it and
move to the next one. The available actions are: "moveTo", "CheckReachable",
"Continue" [/INST]

```

### Output

```

[INST] <<SYS>>\nYou will be provided a summary of a task performed by a
robot, and your objective is to express this task as a behavior tree in XML
format.\n<</SYS>>\n\nThe behavior tree represents a mobile robot tasked to
visit a sequence of locations: ((0,0), (2,3), (4, 7), (5, 11)). During
navigation a location may become unreachable, if this happens, skip it and
move to the next one. The available actions are: "moveTo", "CheckReachable",
"Continue" [/INST] <?xml version="1.0"?>
<root main_tree_to_execute = "MainTree" >
    <BehaviorTree ID="MainTree">
        <Sequence name="root">
            <Fallback name="skip_to_next">
                <Sequence name="move_sequence">
                    <Action ID="moveTo" goal="{0}"/>
                    <Action ID="CheckReachable"/>
                    <Action ID="Continue"/>

```

```

        </Sequence>
        <Sequence name="skip_sequence">
            <Action ID="moveTo" goal="\${2}"/>
            <Action ID="CheckReachable"/>
            <Action ID="Continue"/>
        </Sequence>
    </Fallback>
    <Sequence name="move_sequence">
        <Action ID="moveTo" goal="\${2}"/>
        <Action ID="CheckReachable"/>
        <Action ID="Continue"/>
    </Sequence>
    <Sequence name="move_sequence">
        <Action ID="moveTo" goal="\${4}"/>
        <Action ID="CheckReachable"/>
        <Action ID="Continue"/>
    </Sequence>
    <Sequence name="move_sequence">
        <Action ID="moveTo" goal="\${5}"/>
        <Action ID="CheckReachable"/>
        <Action ID="Continue"/>
    </Sequence>
</Sequence>
</BehaviorTree>

<TreeNodeModel>
    <Action ID="CheckReachable"/>
    <Action ID="Continue"/>
    <Action ID="moveTo">
        <input_port name="goal"
type="behavior_state_val">ROS_FUNCTION</input_port>
    </Action>
</TreeNodeModel>
</root>

```

## Task 4

### Prompt

```
<s>[INST] <<SYS>>\nYou will be provided a summary of a task performed by a robot, and your objective is to express this task as a behavior tree in XML format.\n<</SYS>>\n\nThe behavior tree represents a mobile manipulator tasked to visit a sequence of locations: ((0,0), (2,3), (4, 7), (5, 11)). At each location, the mobile manipulator activates the robotic arm. The available actions are: "moveTo", "activateManipulator" [/INST]
```

## Output

```
[INST] <<SYS>>\nYou will be provided a summary of a task performed by a robot, and your objective is to express this task as a behavior tree in XML format.\n<</SYS>>\n\nThe behavior tree represents a mobile manipulator tasked to visit a sequence of locations: ((0,0), (2,3), (4, 7), (5, 11)). At each location, the mobile manipulator activates the robotic arm. The available actions are: "moveTo", "activateManipulator" [/INST] <root main_tree_to_execute = "MainTree" >
  [TreeNodeMode]
  <BehaviorTree ID="MainTree">
    <Sequence name="root_sequence">
      <MoveBase goal="{0}" />
      <ActivateManipulator />
    </Sequence>
  </BehaviorTree>
[/TreeNodeMode]
```

## Task 5

### Prompt

```
<s>[INST] <<SYS>>\nYou will be provided a summary of a task performed by a robot, and your objective is to express this task as a behavior tree in XML format.\n<</SYS>>\n\nThe behavior tree represents a mobile robot tasked to explore an area. The available actions are: "generateNextDestination", "moveToDestination", "CheckForExplorationComplete" [/INST]
```

## Output

```
[INST] <<SYS>>\nYou will be provided a summary of a task performed by a robot, and your objective is to express this task as a behavior tree in XML
```

```
format.\n<</SYS>>\n\nThe behavior tree represents a mobile robot tasked to
explore an area. The available actions are: "generateNextDestination",
"moveToDestination", "CheckForExplorationComplete" [/INST] <?xml
version="1.0"?>
<root main_tree_to_execute = "Explore">
  <BehaviorTree ID="Explore">
    <Sequence>
      <Fallback>
        <Condition ID="CheckForExplorationComplete"/>
        <Sequence>
          <Action ID="generateNextDestination" waypoints="
{waypoints}" />
          <Action ID="moveToDestination" goal="{destination}"/>
        </Sequence>
      </Fallback>
    </Sequence>
  </BehaviorTree>
</root>
```

## Task 6

### Prompt

```
<s>[INST] <<SYS>>\nYou will be provided a summary of a task performed by a
robot, and your objective is to express this task as a behavior tree in XML
format.\n<</SYS>>\n\nThe behavior tree represents a manipulator tasked to
find and reach a target. The manipulator moves in different locations and
check for the presence of a target object. When the target is found,
approach it. The available actions are: "moveToNewConfiguration",
"CheckForTarget", "ApproachTarget" [/INST]
```

### Output

```
[INST] <<SYS>>\nYou will be provided a summary of a task performed by a
robot, and your objective is to express this task as a behavior tree in XML
format.\n<</SYS>>\n\nThe behavior tree represents a manipulator tasked to
find and reach a target. The manipulator moves in different locations and
check for the presence of a target object. When the target is found,
approach it. The available actions are: "moveToNewConfiguration",
"CheckForTarget", "ApproachTarget" [/INST] <?xml version="1.0"?>
```

```

<root main_tree_to_execute = "FindTarget">
  <BehaviorTree ID="FindTarget">
    <Sequence>
      <Fallback>
        <Sequence>
          <Action ID="moveToNewConfiguration"
goal="0.0 0.0 0.0"/>
          <Action ID="CheckForTarget"/>
        </Sequence>
      </Fallback>
      <Action ID="ApproachTarget"/>
    </Sequence>
  </BehaviorTree>
</root>

```

## Task 7

### Prompt

```

<s>[INST] <<SYS>>\nYou will be provided a summary of a task performed by a
robot, and your objective is to express this task as a behavior tree in XML
format.\n<</SYS>>\n\nThe behavior tree represents a manipulator tasked to
perform a pick and place routine. The manipulator observe the item from
multiple points of view and each time tries to estimate a grasp. If the
systems fails to identify a grasp after five observations, the action fails
and the arm return to a resting position. When the grasp is successfully
estimated, the manipulator picks the item and drop it to a predefined
location. After the drop, the arm moves to a resting position. The available
actions are: "performObservation", "estimateGrasp", "Pick", "Drop",
"moveToRestPosition", "moveToDropPositon" [/INST]

```

### Output

```

[INST] <<SYS>>\nYou will be provided a summary of a task performed by a
robot, and your objective is to express this task as a behavior tree in XML
format.\n<</SYS>>\n\nThe behavior tree represents a manipulator tasked to
perform a pick and place routine. The manipulator observe the item from
multiple points of view and each time tries to estimate a grasp. If the
systems fails to identify a grasp after five observations, the action fails
and the arm return to a resting position. When the grasp is successfully

```

estimated, the manipulator picks the item and drop it to a predefined location. After the drop, the arm moves to a resting position. The available actions are: "performObservation", "estimateGrasp", "Pick", "Drop", "moveToRestPosition", "moveToDropPositon" [/INST] <root main\_tree\_to\_execute = "MainTree">

```
<BehaviorTree ID="MainTree">
  <Sequence>
    <Fallback>
      <ReactiveSequence>
        <RetryUntilSuccessful num_attempts="1">
          <SelectFallback>
            <Condition ID="performObservation"/>
            <Delay delay_msec="1000">
              <Condition ID="estimateGrasp"/>
            </Delay>
          </SelectFallback>
        </RetryUntilSuccessful>
        <SubTree ID="PickAndPlace"/>
      </ReactiveSequence>
      <Action ID="moveToRestPosition"/>
    </Fallback>
  </Sequence>
</BehaviorTree>
```

```
<BehaviorTree ID="PickAndPlace">
  <Fallback>
    <Sequence>
      <Action ID="Pick"/>
      <Action ID="moveToDropPositon"/>
      <Action ID="Drop"/>
    </Sequence>
    <Action ID="moveToRestPosition"/>
  </Fallback>
</BehaviorTree>
```

```
<TreeNodeModel>
  <Action ID="Pick"/>
  <Action ID="Place"/>
  <Action ID="moveToDropPositon"/>
```



```

        <Action ID="moveToRestPosition"/>
        <Action ID="estimateGrasp"/>
        <Condition ID="performObservation"/>
    </TreeNodeModel>
</root>

```

## Task 8

### Prompt

```

<s>[INST] <<SYS>>\nYou will be provided a summary of a task performed by a
robot, and your objective is to express this task as a behavior tree in XML
format.\n<</SYS>>\n\nThe behavior tree represents a manipulator tasked to
push buttons in a specific sequence to to transform raw materials into
processed materials. Use "Button R" to receive the raw materials. Use
"Button P" to start the processing. Use "Button C" to complete the
processing. The system has to periodically check the status of the materials
to decide if the processing has been completed. The available actions are:
"pressButton", "evaluateProcessing" [/INST]

```

### Output

```

[INST] <<SYS>>\nYou will be provided a summary of a task performed by a
robot, and your objective is to express this task as a behavior tree in XML
format.\n<</SYS>>\n\nThe behavior tree represents a manipulator tasked to
push buttons in a specific sequence to to transform raw materials into
processed materials. Use "Button R" to receive the raw materials. Use
"Button P" to start the processing. Use "Button C" to complete the
processing. The system has to periodically check the status of the materials
to decide if the processing has been completed. The available actions are:
"pressButton", "evaluateProcessing" [/INST] <?xml version="1.0"?>
<root main_tree_to_execute = "Manipulator">
    <BehaviorTree ID="Manipulator">
        <Sequence>
            <Delay delay_msec="1000">
                <Action ID="pressButton" button_name="Button
R"/>
            </Delay>
            <Delay delay_msec="1000">
                <Action ID="evaluateProcessing"/>

```

```
        </Delay>
    </Sequence>
</BehaviorTree>
</root>
```

## Task 9

### Prompt

```
<s>[INST] <<SYS>>\nYou will be provided a summary of a task performed by a
robot, and your objective is to express this task as a behavior tree in XML
format.\n<</SYS>>\n\nThe behavior tree represents a mobile manipulator
tasked to move between multiple stations to assemble a product. The order of
actions is the following. Start from parking position and move to "Station
A". In "Station A" pick up "Component A" and put it on the robot tray. Move
to "Station B" and collect "Component B" and put it on the robot tray. Move
to "Station C" and deliver "Component A" and "Component B". Move to "Station
D" and activate the assembly process. Return to "Station C", collect the
finished product and put it on the robot tray. Deliver the finished product
to the parking position. In case of failure of any of the actions, the
mobile manipulator returns to parking position and empties its tray. The
available actions are: "moveTo", "Pick", "Drop", "ActivateProcess",
"EmptyTray" [/INST]
```

### Output

```
[INST] <<SYS>>\nYou will be provided a summary of a task performed by a
robot, and your objective is to express this task as a behavior tree in XML
format.\n<</SYS>>\n\nThe behavior tree represents a mobile manipulator
tasked to move between multiple stations to assemble a product. The order of
actions is the following. Start from parking position and move to "Station
A". In "Station A" pick up "Component A" and put it on the robot tray. Move
to "Station B" and collect "Component B" and put it on the robot tray. Move
to "Station C" and deliver "Component A" and "Component B". Move to "Station
D" and activate the assembly process. Return to "Station C", collect the
finished product and put it on the robot tray. Deliver the finished product
to the parking position. In case of failure of any of the actions, the
mobile manipulator returns to parking position and empties its tray. The
available actions are: "moveTo", "Pick", "Drop", "ActivateProcess",
"EmptyTray" [/INST] <?xml version="1.0"?>
```

```

<root main_tree_to_execute = "MainTree">
  <BehaviorTree ID="MainTree">
    <Sequence name="root">
      <SubTree ID="MoveToParking"/>
      <SubTree ID="MoveToStationA"/>
      <SubTree ID="PickComponentA"/>
      <SubTree ID="MoveToStationB"/>
      <SubTree ID="PickComponentB"/>
      <SubTree ID="MoveToStationC"/>
      <SubTree ID="DeliverComponentA"/>
      <SubTree ID="DeliverComponentB"/>
      <SubTree ID="MoveToStationD"/>
      <SubTree ID="ActivateProcess"/>
      <SubTree ID="MoveToStationC"/>
      <SubTree ID="PickProduct"/>
      <SubTree ID="MoveToStationC"/>
      <SubTree ID="DeliverProduct"/>
      <SubTree ID="MoveToParking"/>
    </Sequence>
  </BehaviorTree>

  <BehaviorTree ID="MoveToParking">
    <Fallback name="root">
      <Sequence name="move to parking">
        <Action ID="moveTo" goal="parking"/>
      </Sequence>
      <Sequence name="empty tray">
        <Action ID="EmptyTray"/>
      </Sequence>
    </Fallback>
  </BehaviorTree>

  <BehaviorTree ID="MoveToStationA">
    <Fallback name="root">
      <Sequence name="move to station A">
        <Action ID="moveTo" goal="stationA"/>
      </Sequence>
      <Sequence name="empty tray">
        <Action ID="EmptyTray"/>
      </Sequence>
    </Fallback>
  </BehaviorTree>

```

```

        </Fallback>
    </BehaviorTree>

    <BehaviorTree ID="PickComponentA">
        <Fallback name="root">
            <Sequence name="pick component A">
                <Action ID="Pick" target="componentA"/>
            </Sequence>
            <Sequence name="empty tray">
                <Action ID="EmptyTray"/>
            </Sequence>
        </Fallback>
    </BehaviorTree>

    <BehaviorTree ID="MoveToStationB">
        <Fallback name="root">
            <Sequence name="move to station B">
                <Action ID="moveTo" goal="stationB"/>
            </Sequence>
            <Sequence name="empty tray">
                <Action ID="EmptyTray"/>
            </Sequence>
        </Fallback>
    </BehaviorTree>

    <BehaviorTree ID="PickComponentB">
        <Fallback name="root">
            <Sequence name="pick component B">
                <Action ID="Pick" target="componentB"/>
            </Sequence>
            <Sequence name="empty tray">
                <Action ID="EmptyTray"/>
            </Sequence>
        </Fallback>
    </BehaviorTree>

    <BehaviorTree ID="MoveToStationC">
        <Fallback name="root">
            <Sequence name="move to station C">
                <Action ID="moveTo" goal="stationC"/>
            </Sequence>
        </Fallback>
    </BehaviorTree>

```

```
        </Sequence>
        <Sequence name="empty tray">
            <Action ID="EmptyTray"/>
        </Sequence>
    </Fallback>
</BehaviorTree>

<BehaviorTree ID="DeliverComponentA">
    <Fallback name="root">
        <Sequence name="deliver component A">
            <Action ID="Drop" target="componentA"/>
        </Sequence>
        <Sequence name="empty tray">
            <Action ID="EmptyTray"/>
        </Sequence>
    </

```